Leap Frog - Hopping over Documents and Modalities

Divija Nagaraju*
Language Technologies Institute
Carnegie Mellon University
dnagaraj@andrew.cmu.edu

Gabriel Moreira*
Language Technologies Institute
Carnegie Mellon University
gmoreira@andrew.cmu.edu

Nikhil Madaan*
Electrical and Computer Engineering
Carnegie Mellon University
nmadaan@andrew.cmu.edu

Rohan Panda*
Electrical and Computer Engineering
Carnegie Mellon University
rohanpan@andrew.cmu.edu

Abstract

Data found on the web generally spans numerous domains and is multi-hop in nature. To build reliable robust automated systems, pipelines would be expected to excel in various tasks such as visual representation learning, knowledge aggregation, cross-modality learning, and question answering. Motivated by the challenges faced in these tasks, we develop Leap Frog, a unified model capable of multimodal question answering. Leap Frog is built based on the models and data provided in WebQA, a well-designed dataset that encourages a good balance between visual performance and language understanding. Leap Frog aims at harmoniously utilizing both visual and textual inputs to answer a given textual question. This would consist of on the one hand choosing the right knowledge sources to aggregate and on the other hand, producing a fluent answer using these sources. We plan to evaluate our model using metrics such as the accuracy and fluency of the answers produced. Furthermore, we aim at conducting analytical studies on the usage of different visual and textual encoders, which may help in producing a better representation of the knowledge sources, which can implicitly help in improving the system's overall performance and robustness.

1 Introduction

The web is a multimodal data trove wherein humans find answers to a myriad of questions. Nevertheless, current Question Answering (QA) systems and datasets fall short of mimicking this experience. This hypothesis is bolstered by the fact that current QA models either rely on textual information alone or they first select a single modality to exploit. The lack of unified systems capable of identifying and exploiting relevant knowledge in different modalities may hinder the progress of complex AI reasoning.

For this project, we proposed to work on the WebQA dataset [19], a novel QA benchmark wherein reasoning over and combining multimodal web-like information is crucial in order to attain a good performance. The task is split in two steps. The first consisting of identifying a subset of correct sources from a set of positives and distractors. The second step boils down to generating a correct answer from the latter. Two types of questions are considered: those whose answers are derived from images and those derived from text.

In the remaining sections of this report we proceed as follows. Firstly, we briefly describe the baseline model that has been evaluated. This consists of the entire QA model put forward in the WebQA paper.

* Everyone contributed equally – alphabetical order

Preprint. Under review.

Next, we analyze CLIP [12], in an exploratory attempt of using coordinated representation learning as a improvement on the existing source retrieval. Subsequently, we put forward the performance metrics employed to evaluate this baseline and the results from the empirical evaluation conducted on the features generated by CLIP. Finally, we analyze these results and draw conclusions of how to leverage them in our final model and potential directions we may consider to explore in this project.

2 Related Work

Textual multi-hop question answering requires models to gather information from different parts of a text to answer a question. Most current approaches learn to address this task in an end-to-end way with large transformer networks. A few recent approaches such as [20] explore modifying the attention mechanism in the language model [4] to encode social dynamics as well. After performing an initial feasibility study, we have decided to pursue the former approach for the duration of the course and leave the rest to future work.

While textual Question-Answering (QA) tasks have been extensively studied, we can identify [1] as the seminal paper that introduced Visual QA (VQA), by grounding the questions on visual data, using images from MS COCO [10]. Ever since, several spin-offs of the original dataset have been proposed along with new VQA datasets. VQA v2.0 [6] tries to reduce the effect of language priors by associating each question with complementary images. Another example is VQA-HAT [3], which provides human attention maps that can be used to assess whether the models are reasoning over the correct image regions. The Visual Genome dataset [9] provides dense image annotations from which a VQA task is derived.

If on the one hand VQA attempts to mimic human reasoning, several shortcuts are often adopted that hinder the progress toward open-domain VQA. By using simple questions which involve e.g., counting or object detection, answering the question often times boils down to a classification task. To counteract this, the OK-VQA [11] dataset requires outside knowledge to answer the questions and features open-ended answers, thus involving more complex reasoning. MultimodalQA [15] introduces different sources of information as well, each from a different modality and requires their integration in order to arrive at the correct answer. In [19] however, the authors argue that the template-generated questions in MultimodalQA prevent true multimodal interaction to be necessary when inferring the answer.

3 Dataset Description

The QA pairs in the dataset are organized as image data and text data. One QA example from [19] is provided in Fig. **??**, with correct sources and distractors. Dataset characteristics are shown in Table 2 and are described more thoroughly in the subsections that follow.

3.1 Image Data

Image data consists of 25K image-based questions, each with an average of 1.4 correct visual facts, 15.3 text distractors, and 15.9 visual distractors. The data points belonging to the image data category, avoid questions that a) are simple facts; b) are easily answered by a text-only search; c) are bound to a specific image. Each image is accompanied by a caption that is only to be used to confirm the name or location of the objects depicted.

The authors of [19] employ hard negative mining to incorporate text and image-based hard negatives. For pertinent text sources, they extract relevant passages from Wikipedia based on noun chunks in the question, while limiting overlap to avoid false negatives. For images, they use both the metadata and visual content to find similar images. Additionally, the authors ensure topic diversity by manually selecting seed prompts with which to execute Bing Visual Search API calls while crawling images.

3.2 Textual Data

Textual data consists of 24K text-based questions, each with an average of 2.0 correct text facts, 14.6 text distractors, and 11.6 visual distractors. To enhance the topic diversity of the multi-hop QA

Table 1: Statistics grouped by the modality required to answer the question.

	# of samples		
Modality	Train	Dev	Test
Image	18,954	2,511	3,464
Text	17,812	2,455	4,076

pairs, the authors employ clustering to group similar entities, however, they ensured that text snippets belonging to a particular topic had a low semantic similarity.

Similar to image data, the dataset includes text as well as image distractors. Relevant text distractors are mined passages from Wikipedia that contain noun phrases from the question and have the highest lexical overlap but lack reference to the answer. Image distractors include the images and captions present on the aforementioned Wikipedia pages, again filtered to include only the ones with high lexical overlap.

4 Model Description

When considering the isolated task of source selection, we will benchmark the CLIP model [12], which consists of a multimodal encoder that uses natural language supervision to train a visual model via contrastive learning. We use the 512 dimensional question and source embeddings produced to compute cosine similarities and thus, the suitability of each source for a given question.

For the complete QA system, we consider the model described in [19] as the baseline for our work. It is built upon the Visual-Language Pre-training (VLP) model [21], which is well-suited for vision-language generation, understanding and as an encoder-decoder structure which can be fine-tuned for various downstream tasks. VLP's input consists of an image, a sentence and special tokens like `[CLS]`, `[SEP]`, `[STOP]` for demarcation. The image is processed as N RoIs which are embedded into N lower dimensional feature vectors using a pre-trained Faster R-CNN [14]. The captions are one-hot encoded and then converted to word-embeddings by a trainable embedding layer. The VLP is then tuned using the masked language model task, similar to the method seen in BERT [4].

4.1 Modality Selector

We add an additional component on top of the baseline suggested in [19], which predicts the target source modality using the question as the only input. The modality selector is referred to as manymodal module in this report as the idea is derived from [7]. The basic structure of the module utilizes BERT as its backbone model, followed by a linear layer. We fine-tune the linear layer which is placed on top of the pretrained BERT to take in textual questions as input and generate encoded features which are used to perform a binary classification of predicting whether the Gold sources are images or texts. This module helps in reducing the search space of the modules that follow and it implicitly improves the question answering phase of the system by getting rid of a few distractors.

The manymodal module achieves an impressive accuracy of 99% on the validation set. This brings to attention the fact that the questions in the dataset have sufficient information that could help us predict, quite confidently, what kind of sources we would want to consider to answer the question. We take advantage of this observation by incorporating this module in the beginning stages as shown in figure 3.

4.2 Representation-based Source Retrieval

Instead of addressing the task of source retrieval as in the WebQA paper, or alternatively through the lens of classification, we also explored the use of coordinated representation spaces [2]. For a source s_j to be relevant when answering a question q_i, the latter and the former must have an overlap in meaning. If we establish a link between this concept of meaning and that of a representation space $\mathcal{S}$, the goal is then to find an encoding transformation f such that for a similarity metric $d(\cdot,\cdot) : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ we have that $d(f(q), f(s_i))$ is large for positive sources and small for distractors. This contrastive learning regime is thus tantamount to modeling the sources with a question-based

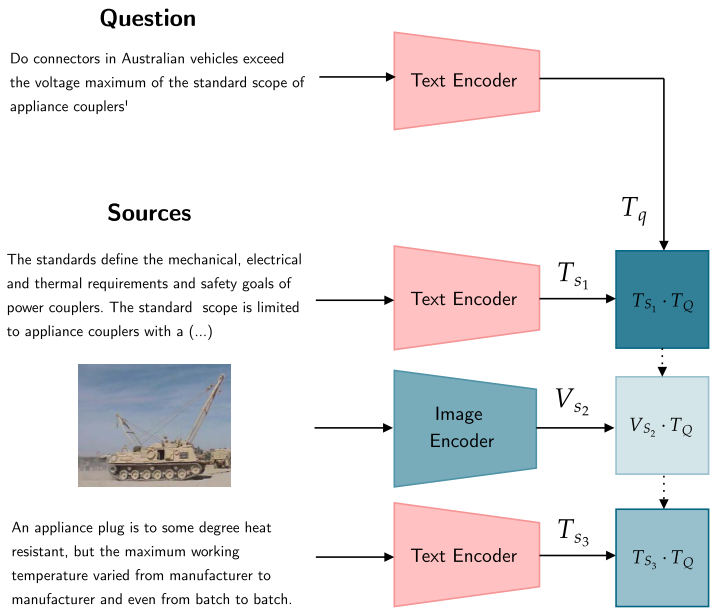


Figure 1: Source retrieval via cosine similarity of question and source embeddings. Question and source examples obtained from the WebQA dataset.

natural language supervision and presents several advantages over the more traditional predictive approach. For one, it is task-agnostic and the representations learned can often be employed in downstream tasks. In addition, these representations are in general richer than those obtained from predictive methods, as reported in [16].

The aforementioned approach can be more formally defined in the context of the WebQA dataset as follows. Given N natural language questions $\{q_i\}_{i=1,\dots,N}$ and a set of M sources $\{s_i\}_{i=1,\dots,M}$ such that $\{s_j\}$, for $j \in \mathcal{G}_i$ correspond to the subset of gold (positive) sources of question q_i, the goal is to find the set of encoder parameters θ such that

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \left\{ -\sum_{i=1}^{N} \frac{1}{|\mathcal{G}_i|} \sum_{j \in \mathcal{G}_i} \log \frac{e^{d\left(f(q_i;\theta), f(s_j;\theta)\right)}}{\sum_{p=1}^{M} e^{d\left(f(q_i;\theta), f(s_p;\theta)\right)}} \right\}, \tag{1}$$

which is just the sum of the crossentropy loss over all questions. Given that questions are textual and sources can be either visual or textual, in practice we must consider two encoders, $f_{\text{text}}(\,\cdot\,;\theta_{\text{text}}) : \{0,1\}^{d_{\text{vocab}} \times l} \to \mathbb{R}^d$ and $f_{\text{visual}}(\,\cdot\,;\theta_{\text{visual}}) : \mathbb{R}^{h \times w \times c} \to \mathbb{R}^d$.

A flagship model that implements a multimodal contrastive learning setup similar to the one described above is CLIP [12]. For a mini-batch containing a set of texts and images, in a one-to-one correspondence, CLIP uses the original Transformer from [17] as the text encoder and either a ResNet [8] or a Vision Transformer (ViT) [5] to encode the images. For each pair image-text, the cosine similarity of the L2-normalized 512-dimensional embeddings is computed, and a loss function similar to (1) is minimized. Given the positive results reported in the paper, especially in zero-shot learning, we decided to benchmark the pre-trained CLIP model in Section 5, as a baseline model for source retrieval. A representation of our approach is shown in Fig. 1.

4.3 Question Answering

In order to generate a natural language answer, MLM loss is used in a seq2seq training regime. The input is formed as `<[CLS],S,[SEP],Q,A,[SEP]>` and uni-directional attention masks are applied to tokens in A to satisfy auto-regressive properties. The decoding then consists of appending the most probable token at a given time-step, followed by a `[MASK]` token. This then forms the input for the

next time step and the decoding continues until [SEP] or [PAD] is generated or the maximum length is reached. Beam search with n=5 is used for the results.

4.4 Evaluation Metrics

The following metrics were used for evaluation of the generated answers.

F1 Score Combines the precision and recall of a retrieval model into a single metric by taking the harmonic mean of precision and recall.

$$F1 - Score = \frac{2 * precision * recall}{precision + recall} \tag{2}$$

Fluency Fluency is measured via BARTScore[18], that is based on the accurate measurement of paraphrase quality. BARTScore(r, c), which can be interpreted as the probability of generating a candidate text (c) given a reference(r). As suggested by the authors of WebQA, since BARTScore doesn't distribute neatly in [0,1] range, we normalize BARTScore(r, c) by the identity score BARTScore(r, r). For the final reference text, we chose the candidate with the maximum BARTScore.

$$FL(c, R) = \max \left\{ \min \left(1, \frac{\text{BARTScore}(r, c)}{\text{BARTScore}(r, r)} \right) \right\}_{r \in R}. \tag{3}$$

Further, this metric prioritizes semantic agreement b) does not heavily punish short sentences (i.e. sentences with < 4 words) as 4-gram BLEU does, c) penalizes word reordering/disfluencies d) and unlike BERTScore, which indiscriminately treats all colors or all shapes as nearly identical, better captures small but critical disparities.

Accuracy The goals of measuring accuracy on WebQA are: 1. Detect the presence of key entities. 2. Penalize the use of any incorrect entities. 3. Avoid penalizing semantically relevant but superfluous words. Visual queries that have closed answer domains (such as color queries, shape queries, and Y/N questions) are evaluated using the F1 score to test for precision. For the remaining visual queries and all textual queries that have diverse and unrestricted answer domains, the probability of cheating by guessing a long list of keywords is small and such answers will be penalized by BARTScore, so we evaluate accuracy for such questions via recall (*RE*). With *c* as a candidate output, *K* for correct answer keywords, and *qc* for question category, *Acc* score is defined as

$$\text{Acc}(c, K) = \begin{cases} F1(c \cap D_{qc}, K \cap D_{qc}) & \text{if } qc \in [\text{color, shape, number, } Y/N] \\ RE(c, K) & \text{otherwise.} \end{cases} \tag{4}$$

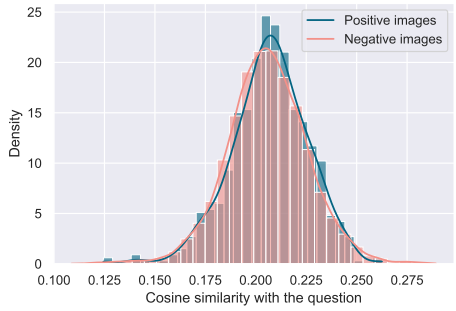
5 Experiments

5.1 Exploratory Data Analysis using CLIP

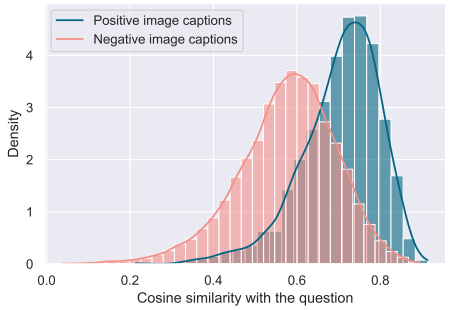
In order to assess the feasibility of using CLIP for source retrieval, we used the pretrained CLIP Transformer and the pretrained CLIP ViT-B/32 available on GitHub to encode the textual and visual information, respectively, in the validation split of WebQA. Since CLIP uses a context size of 77, all texts, including questions, image captions and titles were trimmed to this length. Similarly, in order to fit the 224×224 resolution of the visual encoder, all images were resized. Once we computed the embeddings, an L2-normalization was performed and the inner products were computed, thus obtaining the cosine similarities.

We conducted four different similarity experiments to assess which information was best to discriminate between positive and distractor sources: similarity between the question embedding and that of the source texts, the source images, the source image captions and the source image titles. An histogram of the cosine similarities can be observed in Fig. 2. An initial finding is that the distribution of similarities between negative sources and the question follows, in almost every case, a Gaussian curve which was to be expected. The only exception is Fig. 2d, where we can see two peaks.

(a) Text sources and question.



(b) Image sources and question.



(c) Image captions and question.

(d) Image titles and question.

Figure 2: Cosine similarities between positive and negative source and the question embeddings, computed using CLIP with ViT-B/32 as the visual encoder. Histograms computed for a subset of the validation split containing 10^3 question-sources pairs.

While there is a clear difference between the positive and negative texts (Fig. 2a) distributions, the similarity distribution of positive and negative images completely overlaps (Fig. 2b). The same does not happen when, instead of using the raw image, we use its caption (Fig. 2c) or its title (Fig. 2d). In fact, both these image attributes yield an even better discriminative power than the text sources themselves, the title being the best when it comes to telling apart a positive image from a distractor. Coincidentally, out of all the types of textual sources, the title is often the smallest one, many times consisting of a single word.

Regarding the lackluster performance of CLIP in distinguishing between positive and negative images note that the experiment showed that the image titles and captions are, in general, *similar* to the question. This means the raw image content should be as well. However, the visual encoder is currently not extracting this information. This may be due to the fact that we did not fine-tune CLIP on WebQA, and the images therein may be considerably different from those used to trained the ViT-B/32 visual encoder. In addition, since the captions and titles are encoded with the same Transformer as the question, a better degree of similarity was expected. While these results are still in their early stages, they shed light on two aspects: fine-tuning CLIP on WebQA is worth exploring further; translating all the source modalities to text may yield more accurate and consistent results.

5.2 Baseline Studies

In order to get the baseline scores for the problem, we make use of the VLP weights, shared by [19] for running the baseline model for source retrieval and question answering.

Input Representations All the text-based knowledge sources which include textual sources, image captions, questions, and answers, are tokenized by the Bert-base-cased [4] tokenizer. Images are represented by 100 regions predicted by a variant of Faster RCNN with a ResNeXt-101 FPN backbone, pretrained on Visual Genome.

Source Retrieval For retrieval, candidate sources and questions are fed into the model one by one (<[CLS], S, [SEP], Q, A, [SEP]>), which predicts the probability of selecting a particular source. Let $\mathcal{G}$ and $\mathcal{D}$ denote the set of gold sources and distractors for a sample. The loss function used for retrieval is as follows.

$$\text{Loss}_{\text{retrieval}} = \sum_{s_i \in \mathcal{G}} \log(p_{s_i}) + \sum_{s_i \in \mathcal{D}} \log(1 - p_{s_i}) \tag{5}$$

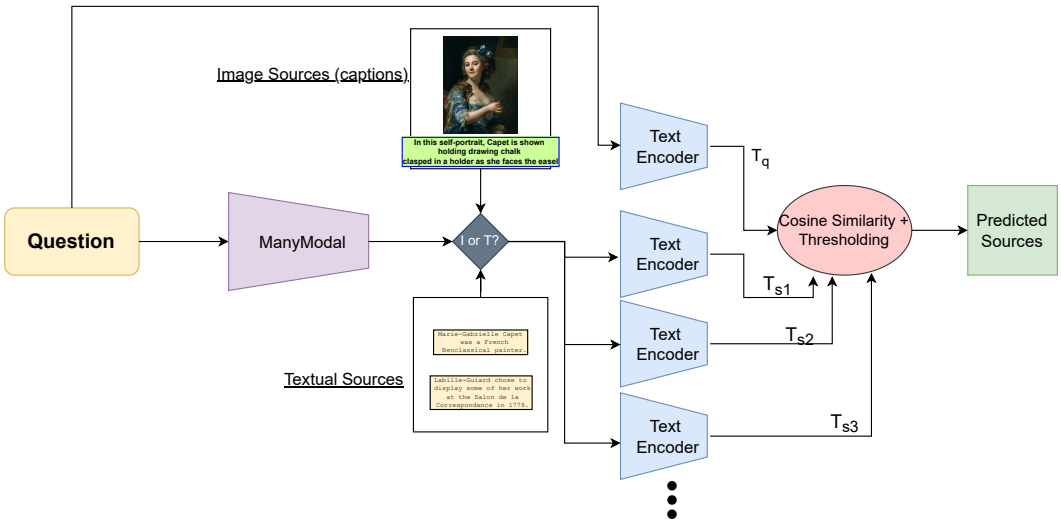


Figure 3: Source retrieval via cosine similarity of question and source embeddings. The modality of the predicted sources are determined using manymodal module.

Question Answering We feed <[CLS], S, [SEP], Q, A, [SEP]> to the model, where masked attention masks are used in order to satisfy the auto-regressive property. For decoding we iteratively append a [MASK] to the end of the input, replacing it with a predicted token and appending a new [MASK] for the next timestep. We stop the decoding process upon seeing [SEP], [PAD], or reaching a maximum length. We use beam search (n=5) and choose the most confident output for evaluation.

In table 2 we present the results from our empirical evaluation of the VLP model from [19] on the validation split.

Table 2: F1, Accuracy and Fluency scores on the validation split of the VLP model.

Baseline Results			
Model	Lexical-F1	FL	Acc
VLP	68.9	41.8	36.54

6 Improvements Over Baseline

6.1 Improving Retrieval Performance

Motivated by the observations explained in section 5.1, we experiment with various competitive encoders that could provide useful representations using only textual data, i.e. questions, textual sources, and image captions(instead of image features), which can be used to retrieve the relevant sources to generate the answer. The overall pipeline for retrieval is shown in figure 3. We compare the performances of BERT [4], sentence transformer [13] and CLIP [12] in providing encoded features that are used for predicting the golden sources in Table 3. The basic steps included in source retrieval consist of encoding the question and the text from textual and image sources (we used image captions)

using an encoder, calculating similarity scores between the encoded questions and sources, and choosing sources based on a binary classification by thresholding the similarity scores. Since [19] also incorporates image features in their retrieval performance evaluation, we do not include their retrieval in our comparisons as a baseline as we wish to explore how far can a model be tuned to perform well only using textual information. We, therefore use CLIP w/o finetuning as our baseline since CLIP is one of the SOTA models used widely in this domain and we used this model for our initial analysis in section 5.1.

Table 3: F1 scores of various models in source retrieval

Models	Retrieval F1
BERT (pre-trained)	18.49
CLIP (pre-trained)	16
Sentence Transformer (pre-trained)	20.2
BERT w Contrastive Loss	45.3

Based on the F1 scores for retrieval on the validation set in table 3, all the models perform as well as, or better than the baseline. In the case of fine-tuned BERT, we see a significant increase of 29.3% from the baseline. However, when the model is not finetuned, the highest increase we could find was around 4.2%, when sentence transformer was used. We can also note that though fine-tuned BERT does not outperform the model proposed in [19], it reaches a reasonable F1 score using just textual information while completely discarding visual information. We believe that further finetuning and training of the model would improve the retrieval performance to reach competitive scores with respect to [19].

7 Conclusion

The baseline studies and improvements allow us to make the following conclusions. Answer modality is highly correlated with the wording of the question. This enables us to add a modality selector component on top of the baseline model and achieve significant improvements. Additionally, we infer that textual information is discriminative enough for source pre-selection.

As future work we suggest exploration of better pre-processing techniques since the dataset contains texts and captions scraped from the web. This makes the data highly heterogeneous and includes languages, characters and symbols unknown to the language models used. During our exploratory data analysis phase we found that images carry valuable information that CLIP could not extract. Improving visual feature extraction through efficient object detection could be another avenue for further studies.

References

[1] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual Question Answering. *Proceedings of the IEEE International Conference on Computer Vision*, 2015 International Conference on Computer Vision, ICCV 2015:2425–2433, 2015.

[2] Tadas Baltrusaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2):423–443, 2018.

[3] Abhishek Das, Harsh Agrawal, C Lawrence Zitnick, Devi Parikh, and Dhruv Batra. Human attention in visual question answering : Do humans and deep Networks look at the same regions ? 2015.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[6] Yash Goyal, Tejas Khot, Aishwarya Agrawal, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: Elevating the role of image understanding in visual question answering. *International Journal of Computer Vision*, 127(4):398–414, 2019.

[7] Darryl Hannan, Akshay Jain, and Mohit Bansal. Manymodalqa: Modality disambiguation and qa over diverse inputs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7879–7886, 2020.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[9] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017.

[10] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision*, pages 740–755. Springer, 2014.

[11] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. OK-VQA: A Visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3195–3204, 2019.

[12] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[13] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

[14] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal network. *Advances in neural information processing systems*, 28, 2015.

[15] Alon Talmor, Ori Yoran, Amnon Catav, Dan Lahav, Yizhong Wang, Akari Asai, Gabriel Ilharco, Hannaneh Hajishirzi, and Jonathan Berant. MultimodalQA: Complex question answering over text, tables and images. *arXiv preprint arXiv:2104.06039*, 2021.

[16] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *European conference on computer vision*, pages 776–794. Springer, 2020.

[17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[18] Yuan Weizhe, Graham Neubig Neubig, and Pengfei Liu. Bartscore: Evaluating generated text as text generation. *arXiv preprint arXiv:2106.11520*, 2021.

[19] Chang Yinghsan, Narang Mridu, Suzuki Hisami, Cao Guihong, Gao Jianfeng, and Bisk Yonatan. WebQA: Multihop and Multimodal QA. 2021.

[20] Amir Zadeh, Michael Chan, Paul Pu Liang, Edmund Tong, and Louis-Philippe Morency. Social-IQ: A Question Answering Benchmark for Artificial Social Intelligence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[21] Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason Corso, and Jianfeng Gao. Unified vision-language pre-training for image captioning and VQA. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13041–13049, 2020.